

First Experimental Assessment of ABNO-driven In-Operation Flexgrid Network Re-Optimization

Ll. Gifre, F. Paolucci, L. Velasco, A. Aguado, F. Cugini, P. Castoldi, and V. López

Abstract—Traffic affected by link failures can be recovered using path restoration schemes. In dynamically operated networks provided with a control plane, restoration algorithms run in a centralized element, such as the Path Computation Element (PCE). To increase traffic restorability in flexgrid networks, multiple paths, *sub-connections*, can be used to restore every single affected connection. However, the multipath restoration scheme might result in a poor resource utilization entailing a lesser grade of service. In-operation network planning algorithms can be used to mitigate this problem once the failed link is repaired; we propose solving the so-called multipath after failure repair optimization problem (MP-AFRO) to reduce sub-connections count by aggregating those belonging to the same original connection and re-routing the resulting connection to release spectral resources. The MP-AFRO problem is modelled using a Mixed Integer Linear Program (MILP) formulation. In view of the complexity of the model and the limited time to solve the problem, we propose a heuristic algorithm that provides a good trade-off between complexity and optimality. The performance on the MP-AFRO heuristic is firstly validated by simulation. Next, the heuristic algorithm is deployed inside an in-operation planning tool in the form of back-end PCE (bPCE) inside the Application-based Network Operations (ABNO) architecture controlling a network; the bPCE is connected to the centralized active stateful PCE. MP-AFRO is experimentally demonstrated using a distributed field trial test-bed connecting the premises of Telefonica (Madrid), CNIT (Pisa), and UPC (Barcelona).

Index Terms—Network Experiments, Flexgrid Planning, In-operation Planning.

I. INTRODUCTION

In-operation network planning consists in making network resources available by reconfiguring and/or re-optimizing the network on demand and in real-time [1]. Although the concept of in-operation planning is general and can be applied to any network, in this work we focus on flexgrid-based optical core networks. The flexgrid technology split the optical spectrum into multiple equally-sized frequency *slices*, where a variable number of consecutive slices (*frequency slot*) can be allocated to each connection depending on the requested bandwidth and the used modulation format.

To automate connection provisioning, a control plane architecture is needed. To that end, the Path Computation Element (PCE) [2] or a software-defined networking (SDN) controller [3] can be deployed. In flexgrid networks, path computation consists in solving the routing and spectrum allocation (RSA) problem [4], which involves two different constraints: the *continuity constraint* to ensure that the

allocated slot is the same along the links in the route and the *contiguity constraint* to guarantee that the slices defining the slot are contiguous in the spectrum.

Some previous works have experimentally demonstrated in-operation flexgrid network planning use cases, such as spectrum defragmentation, using a SDN controller [5] or a PCE [6]. Notwithstanding, the IETF is currently defining the Application-Based Network Operations (ABNO) architecture [7] to implement complex workflows and compute time consuming optimization algorithms. ABNO is based on functional elements, such as the active stateful PCE [8], [9]. To solve optimization problems, the front-end/back-end (fPCE/bPCE) architecture can be used, where the bPCE is an in-operation planning tool [10], whilst the fPCE performs simple path computations and delegates complex ones to the bPCE. The PCE Protocol (PCEP) [11] can be used as a request/response protocol among ABNO modules. In addition, both the Traffic Engineering Database (TED) and the Label Switched Path Database (LSP-DB) need to be synchronized between fPCE and bPCE; the BGP-LS protocol, currently being standardized [12], can be used to synchronize the TED, whereas LSP-DB synchronization can be implemented by using PCEP PCRpt messages [13].

A first implementation of ABNO controlling flexgrid networks was shown in [14]. Authors in [15] demonstrated the spectrum defragmentation use case using ABNO, where the central frequency of some existing optical connections is shifted to make enough room to serve a new connection request. As a different use case, in this paper we face the after failure repair optimization (AFRO) problem, defined in [16]. When a fiber cut occurs, connections affected are restored; when the link is repaired not only those connections that were restored, but also any other connection that might use the repaired link, can be re-routed.

Specifically in flexgrid networks, recovery schemes need to compute a new RSA for every connection affected by a failure. Since connection restorability is related to the availability of continuous and contiguous spectrum resources, the multipath (MP) restoration scheme was proposed in [17] to increase restorability; it divides original connections into several parallel disaggregated connections (*sub-connections*), which are routed independently among them, additionally enabling recovering only part of the original bitrate (*bitrate squeezing*). Although multipath restoration has benefits, it provides poor resource utilization and is spectrally inefficient and hence, re-optimizing the

Manuscript received June 7, 2014.

Ll. Gifre (lgifre@ac.upc.edu) and L. Velasco are with the Optical Communications Group (GCO) at Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.

F. Paolucci, F. Cugini, and P. Castoldi are with Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT), Pisa, Italy.

A. Aguado and V. López are with Telefónica Investigación y Desarrollo (TID), Madrid, Spain.

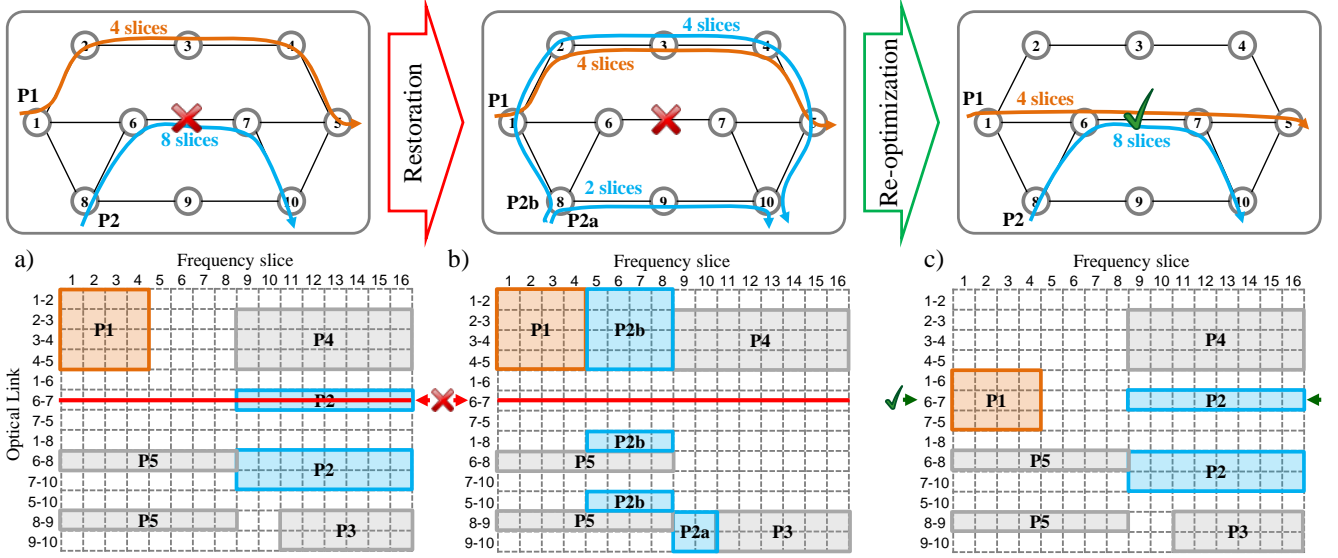


Fig. 1. An example of multi-path restoration and after failure repair optimization. 6.25 GHz frequency slices are used.

network just after a link is repaired becomes essential. To that end, the AFRO problem can be enhanced with the capability of merging sub-connections and increasing the served bitrate to its original value; we call it as MP-AFRO.

This paper extends the work presented in [18]. The contribution is three-fold: *i)* firstly a Mixed Integer Linear Program (MILP) formulation for the MP-AFRO problem is devised. In view of its computational complexity, a heuristic algorithm able to provide a better trade-off between complexity and optimality is proposed; *ii)* next, after checking the feasibility of PCEP to encode MP-AFRO computation requests as well as their solutions, a workflow to implement MP-AFRO in the ABNO architecture is proposed; *iii)* finally, the proposed workflow and the heuristic algorithm are experimentally validated over a distributed field trial set-up connecting premises in Telefonica (Madrid, Spain), CNIT (Pisa, Italy), and UPC (Barcelona, Spain).

The remaining of this paper is organized as follows. Section II gives an overview to MP recovery and AFRO. Section III defines the MP-AFRO problem and presents two solving methods: a MILP formulation and a heuristic algorithm to solve MP-AFRO for real-world scenarios. Section IV describes the considered workflow for the MP-AFRO algorithm. Next, section V validates the MP-AFRO heuristic using simulation and presents the distributed experimental test-bed where MP-AFRO has been assessed. Finally, section VI concludes the paper.

II. RESTORATION AND AFTER FAILURE REPAIR OPTIMIZATION

For illustrative purposes, Fig. 1 reproduces a small flexgrid network topology where several connections are currently established; in particular, the route of connections *P1* and *P2* is shown. The spectrum usage is also provided in Fig. 1, where the spectrum allocation for all five established connections is specified. Three snapshots with the state of the network are shown: the link 6-7 has failed in Fig. 1a; multipath restoration has been applied in Fig. 1b, and connection *P2* has been split into two parallel sub-connections *P2a* and *P2b* squeezing the total conveyed bitrate to fit into the available free spectrum slots; failed link 6-7 has

been repaired in Fig. 1c, and re-optimization has been performed by solving MP-AFRO, so that sub-connections have been merged back and bitrates have been expanded to its originally requested values.

In view of the example, it is clear that multipath restoration allows increasing restorability, in particular when no enough contiguous spectrum can be found along a single path, as happened when restoring *P2*. Nonetheless, this benefit is at the cost of an increased resource usage, not only as a result of using (not shortest) parallel routes, and squeezing the total conveyed connection's bitrate, but also because the spectral efficiency is degraded when connections are split. For instance, a 400 Gb/s aggregated flow can be conveyed on one single 100 GHz connection or on four parallel 37.5 GHz sub-connections, therefore using 50% more spectral resources even in the case of being routed through same links.

For this very reason, resource utilization can be improved by applying MP-AFRO, i.e. by re-routing established connections on shorter routes, by merging parallel sub-connections to achieve better spectrum efficiency, and expanding the conveyed connections' bitrates to its original ones. Fig. 1c illustrates an example of such re-optimization, where connection *P1* has been rerouted using a shorter route that includes the repaired link, whilst sub-connections *P2a* and *P2b* have been merged on a single connection conveying the originally requested bandwidth. Note however that although those operations might entail short traffic disruption, it can be minimized using the standardized *make-before-break* rerouting technique [19], as revealed in [16].

III. AFTER FAILURE REPAIR OPTIMIZATION WITH MULTI-PATH MERGING (MP-AFRO)

In this section the MP-AFRO problem is formally stated and a MILP formulation to solve it is presented. After analyzing the complexity of the MILP model, a heuristic algorithm is presented to provide near-optimal solutions in the required computation times.

A. Problem Statement

The MP-AFRO problem can be formally stated as follows:

- *Given:* a) a network topology $G(N, E)$ defined as a set of

optical nodes N and a set of fiber links E ; b) an optical spectrum divided into frequency slices of a given width; c) the set of slices used by non-selected connections; d) a set of demands candidate for re-optimization, D .

- *Find*: the route and spectrum allocation for demands in D , merging sub-connections serving each original connection.
- *Objective*: maximize the bitrate served while minimizing the total number of sub-connections used to convey the traffic served.

An MILP-based model is presented in next section.

B. MILP Formulation

The MILP model is based in a *link-path* formulation for RSA [4] where a set of routes is computed for each of the candidate demands to be re-optimized.

The following sets and parameters have been defined:

E	Set of network links, index e .
S	Set of slices in the spectrum of each link, index s .
D	Set of candidate demands, index d . Each demand d represented by tuple $\langle s_d, t_d, b_d \rangle$, where s_d is the source node, t_d is the target node, and b_d is the requested bitrate.
$P(d)$	Set of sub-connections being used to serve d .
$K(d)$	Set of pre-computed routes for demand d , index k .
$C(d)$	Set of slots for demand d , index c .
R	Set of pairs reach-modulation format, index r .
δ_{ke}	1 if route k contains link e ; 0 otherwise.
α_{es}	1 if slice s in link e is used by any already established connection not in D ; 0 otherwise.
γ_{cs}	1 if slot c contains slice s ; 0 otherwise.
β	Objective function weight.
$b(c, r)$	Maximum bitrate that can be conveyed using slot c with the modulation format given by pair r .
$b(k, c)$	Maximum bitrate that can be conveyed through route k using slot c .
$len(r)$	Reachability of pair r (in Km).
$len(k)$	Length of route k (in Km).

The decision variables are:

x_{dkc}	Binary. 1 if route k and slot c are used to serve demand d ; 0 otherwise.
y_{dkr}	Binary. 1 if pair r is used for demand d in route k ; 0 otherwise.
z_{dkc}	Positive Real. Served bitrate for demand d through route k and slot c .
w_d	Positive Integer. Total number of sub-connections used to serve demand d .

The $K(d)$ set is computed using eq. (1) as the union between the already-in-use paths $P(d)$ and the set of all shortest paths between s_d and t_d of the same length in hops as the shortest one, using the repaired link e .

$$K(d) = P(d) \cup \left\{ r \in KSP(s_d, t_d), e \in r \wedge |r| = |SP(s_d, t_d)| \right\} \quad \forall d \in D \quad (1)$$

The MILP formulation for the MP-AFRO is as follows:

$$\text{Max } \Phi = \sum_{d \in D} \left(\frac{\beta}{b_d} \sum_{k \in K(d)} \sum_{c \in C(d)} z_{dkc} - \frac{1}{|P(d)|} w_d \right) \quad (2)$$

subject to:

$$\sum_{k \in K(d)} \sum_{c \in C(d)} x_{dkc} = w_d \quad \forall d \in D \quad (3)$$

$$\sum_{d \in D} \sum_{k \in K(d)} \sum_{c \in C(d)} \delta_{ke} \cdot \gamma_{cs} \cdot x_{dkc} \leq (1 - \alpha_{es}) \quad \forall e \in E, s \in S \quad (4)$$

$$w_d \leq |P(d)| \quad \forall d \in D \quad (5)$$

$$\sum_{k \in K(d)} \sum_{c \in C(d)} z_{dkc} \leq b_d \quad \forall d \in D \quad (6)$$

$$z_{dkc} \leq b_d \cdot x_{dkc} \quad \forall d \in D, k \in K(d), c \in C(d) \quad (7)$$

$$z_{dkc} \leq \sum_{r \in R} b(c, r) \cdot y_{dkr} \quad \forall d \in D, k \in K(d), c \in C(d) \quad (8)$$

$$\sum_{c \in C(d)} len(k) \cdot x_{dkc} \leq \sum_{r \in R} len(r) \cdot y_{dkr} \quad \forall d \in D, k \in K(d) \quad (9)$$

$$\sum_{r \in R} y_{dkr} = 1 \quad \forall d \in D, k \in K(d) \quad (10)$$

Objective function in eq. (2) maximizes the total served bitrate while minimizing the amount of sub-connections used to serve that bitrate. Constraint (3) accounts for the number of sub-connections used to serve each demand. Constraint (4) ensures that any single slice is used to convey one sub-connection at last, provided that it is not already used by other demand not in D . Constraint (5) guarantees that sub-connections count to serve any specific demand is not increased. Constraint (6) assures that served bitrate does not exceeds demand's requested bitrate. Constraint (7) sets to zero bitrate of unused sub-connections. Constraint (8) limits the bitrate conveyed by any specific sub-connection to the maximum associated to pair r , whereas constraint (9) limits the length of each used sub-connection to the reachability associated to pair r . Finally, constraint (10) selects one pair r for each sub-connections.

Regarding complexity, the MP-AFRO problem is NP-hard since simpler network routing problems have been proved to be NP-hard (see e.g. [20]). Regarding its size, the number of variables is $O(|D| \cdot |K(d)| \cdot (|C(d)| + |R|))$ and the number of constraints is $O(|D| \cdot |K(d)| \cdot |C(d)| + |E| \cdot |S|)$.

Although the MILP model can be solved in a short period of time, e.g. minutes using a standard solver such as CPLEX or dozens of seconds if a column generation algorithm is used [16], during MP-AFRO computation new connections could arrive, which need to be queued and delayed until the MP-AFRO solution is implemented in the network. Aiming at reducing provisioning delay, providing a good trade-off between complexity and optimality, a heuristic algorithm is presented next to solve the MP-AFRO in the stringent required times (e.g. < 1 s.).

C. Heuristic Algorithm

The MP-AFRO algorithm is shown in Table I. The algorithm maximizes the amount of bitrate that is served (note that only part of the original bitrate could be restored), while minimizing the number of sub-connections per demand being re-optimized.

TABLE I MP-AFRO HEURISTIC ALGORITHM

IN: $G(N, E), D, \text{maxIter}$
OUT: bestS
1: $\text{bestS} \leftarrow D$
2: for each d in D do deallocate($G, d.SC$)
3: for $i = 1..\text{maxIter}$ do
4: $S \leftarrow \emptyset; G' \leftarrow G; \text{feasible} \leftarrow \text{true}$
5: sort(D , random)
6: for each d in D do
7: $d.SC' \leftarrow \emptyset; d.\text{reoptBw} \leftarrow 0$
8: $\{<k, c_k>\} \leftarrow \text{getMP_RSA}(d, d.\text{servedBw})$
9: if $b(\{<k, c_k>\}) < d.\text{servedBw}$ then
10: $\text{feasible} \leftarrow \text{false}; \text{break}$
11: $d.\text{reoptBw} \leftarrow b(\{<k, c_k>\}); d.SC' \leftarrow \{<k, c_k>\}$
12: allocate($G', d.SC'$); $S \leftarrow S \cup \{d\}$
13: if not feasible then
14: for each d in S do deallocate($G', d.SC'$)
15: continue
16: for each d in D do
17: if $d.\text{reoptBw} < d.bw$ then
18: $\{<k, c_k>\} \leftarrow \text{expandSlots}(d, d.bw, G')$
19: if $b(\{<k, c_k>\}) > d.\text{reoptBw}$ then
20: $d.\text{reoptBw} \leftarrow b(\{<k, c_k>\})$
21: $d.SC' \leftarrow \{<k, c_k>\}; \text{continue}$
22: if $d.\text{reoptBw} < d.bw \ \& \ d.SC' < d.SC $ then
23: $\{<k, c_k>\} \leftarrow \text{getMP_RSA}(d, d.bw - d.\text{reoptBw})$
24: if $b(\{<k, c_k>\}) = 0$ then break
25: $d.\text{reoptBw} \leftarrow d.\text{reoptBw} + b(\{<k, c_k>\})$
26: $d.SC' \leftarrow d.SC' \cup \{<k, c_k>\}$
27: allocate($G', d.SC'$)
28: Compute $\Phi(S)$
29: for each d in D do deallocate($G', d.SC'$)
30: if $\Phi(S) > \Phi(\text{bestS})$ then $\text{bestS} \leftarrow S$
31: for each d in D do allocate($G, d.SC$)
32: return bestS

The algorithm receives as input the network's TED, i.e. $G(N, E)$, the candidate list of demands to be re-optimized D , and the maximum number of iterations maxIter ; it returns the ordered set bestS containing the best solution found, where the list of sub-connections to serve each demand is specified.

All the sub-connections of each demand are first deallocated from the original TED in G (line 2 in Table I) and then, the algorithm performs a number of iterations (maxIter), where the set D is served in a random order (lines 3-5). At each iteration, a copy of the original TED is stored into an auxiliary TED G' , so that every subsequent operation is performed over G' . Each iteration consists of two steps performed sequentially. In the first step (lines 6-12), the set of demands is served ensuring the currently served bitrate in the hope of finding a shorter route. The getMP_RSA function computes the set of sub-connections consisting of the routes with the highest available capacity, using eq. (11), and minimum cost and selects those to serve the required bitrate (line 8). A solution is feasible only if every demand obtain at least the same bitrate than the current allocation so in case the required bitrate cannot be served, the solution is discarded (lines 9-10), otherwise the resources selected are reserved in the auxiliary TED G' (line 12). In the second step, the bitrate of the demands is increased firstly by allocating wider slots along the same routes (lines 17-21) and then by adding more sub-connections.

$$c_k(d) = \begin{cases} c^* & \left| \begin{array}{l} b(k, c^*) \geq b(k, c) \ \forall c \in C(d) \\ \alpha_{es} = 0 \ \forall e \in k, s \in S(c) \end{array} \right\} \end{cases} \quad (11)$$

After a complete solution is obtained, its fitness value is computed (line 28). Next, the used resources in the solution are released from the auxiliary TED (line 29). The fitness value of the just obtained solution is compared to that of the best solution found so far (line 30) and stored provided that it is feasible and its fitness value is better than that of the incumbent. The state of G is restored and the best solution is eventually returned (lines 31-32).

In the next section, we focus on devising the workflow that needs to be carried out in the control plane to be able to request solving the MP-AFRO problem and deploy the obtained solution.

IV. PROPOSED RE-OPTIMIZATION WORKFLOW

To deal with network re-optimization, we consider a control plane based on the ABNO architecture, which includes an fPCE responsible for computing provisioning requests and dealing with network data plane, and a bPCE capable of performing complex computations to solve optimization problems.

We assume that an operator in the Network Management System (NMS) triggers the MP-AFRO workflow after a link has been repaired. To that end, the NMS issues a service request towards the ABNO Controller. Fig. 2 reproduces the re-optimization sequence diagram (Fig. 2a) involving ABNO components and its execution flow diagram (Fig. 2b). When the request from the NMS arrives at the ABNO controller, re-optimization is requested by sending a PCReq message (labeled as 1 in Fig. 2) to the fPCE. Upon receiving the request, the fPCE collects relevant data to be sent to the bPCE in the form of a PCReq message containing a Global Concurrent Optimization (GCO) request (2).

In light of the MP-AFRO problem statement, and assuming that the network topology and the current state of the resources has been synchronized, the information to be included in the GCO request is the set of connections candidate for re-optimization, D . Therefore, an algorithm to find the candidate connections in the LSP-DB is needed. The

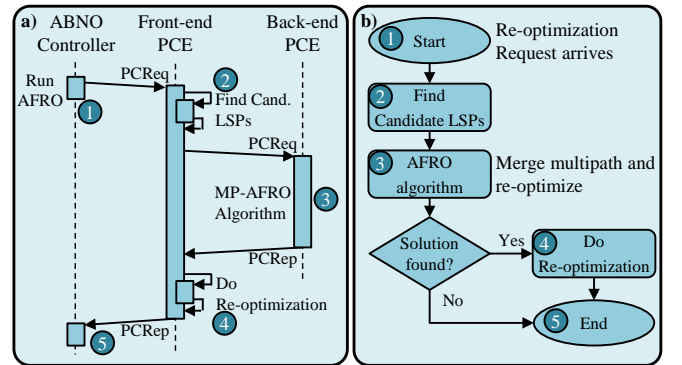


Fig. 2. Re-optimization sequence (a) and flow (b) diagrams

TABLE II FIND CANDIDATE DEMANDS ALGORITHM

IN: $G(N, E), L, e$
OUT: D
1: $D \leftarrow \emptyset$
2: for each $d \in L$ do
3: $\{SP\} \leftarrow \text{KSP}(d.s, d.t, G)$
4: for each $SP \in \{SP\}$ do
5: if $e \in SP$ then $D \leftarrow D \cup \{d\}$
6: return D

algorithm, presented in Table II, receives as input the network's TED in G , the set L with the LSP-DB, and the repaired link e . The connections whose shortest path traverses the repaired link are selected and the candidate list of demands D to be re-optimized is eventually returned.

Information regarding the candidate connections is sent in a PCReq message containing the GCO request. Each candidate connection is sent as an individual request, identified by a RP object; the end points and original bandwidth are also included using the END-POINTS and the BANDWIDTH objects, respectively. In addition, for each individual sub-connection related to the original connection, its current route and spectrum allocation are specified using a RRO (Record Route Object) object. The RRO object interleaves the route's nodes with its spectrum allocation. The sequence of nodes is encoded using the Unnumbered Interface sub-objects encoding each node and the corresponding outgoing port identifiers to reach the next node; and an IPv4 Prefix sub-object to identify the destination node.

The spectrum allocation is encoded using the Label Control sub-object that contains the tuple $\{n, m\}$ that unambiguously defines any frequency slot, where n is the central frequency index (positive, negative or 0) of the selected frequency slot from a reference frequency (193.1 THz), whereas m is the slot width in number of slices at each side of the central frequency. Aiming at finding an optimal solution for the entire problem, individual requests are grouped together using a SVEC object.

The desired network-wide GCO related criterion, such as "MP-AFRO", is specified by means of the OF object. Finally, the repaired link should/must be used for re-optimization. To that end, we extended current standards adding an IRO (Include Route Object) object that identifies those repaired link that should be included into the new routes in a symmetric way with respect to the XRO (Exclude Route Object) object that specifies the links that should be excluded from routed being computed.

Upon receiving the PCReq, the bPCE runs the specified heuristic algorithm. The solution of the MP-AFRO problem is encoded in a PCRep (labeled as 3 in Fig. 2); each individual request is replied specifying the bitrate that could be served and the route and spectrum allocation of the connections related to each request in a list of ERO objects. Note that the solution might entail merging several existing sub-connections to create one or more new connections. The Order TLV is included in RP objects to indicate the order in which the solution needs to be implemented in the data plane.

Upon receiving the PCRep message with the solution from the bPCE, the fPCE runs the *Do-Optimization* algorithm (labeled as 4 in Fig. 2), listed in Table III, to update the connections in the network's data plane. The algorithm sorts D in the same order than S (line 1 in Table III). Next, each demand d in D is sequentially processed. The possibly-updated demand ds in S corresponding to the current demand d in D is taken (line 3) by matching them using the identifier stored in its related RP object. If d differs from ds , the bPCE has found a better set of sub-connections for the demand, so the fPCE should send the corresponding PCUpd/PCInit

TABLE III DO RE-OPTIMIZATION ALGORITHM

IN: $D, S, order$	
1:	sort($D, S, order$)
2:	for each d in D do
3:	$ds \leftarrow \text{getDemand}(S, d)$
4:	if $d \neq ds$ then update(d, ds)

TABLE IV BITRATE-SPECTRUM WIDTH

Bitrate (Gb/s)	Bandwidth (GHz)	#Slices
100	37.5	6
200	62.5	10
400	100	16

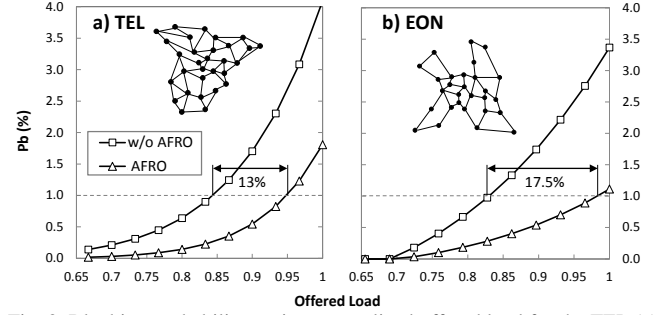


Fig. 3. Blocking probability against normalized offered load for the TEL (a) and EON (b) network topologies.

messages towards source GMPLS controllers (line 4) to update/implement it [8], [9].

When the solution has been completely implemented i.e., when the corresponding PCRpt messages are received, the fPCE replies the completion of the requested operation to the ABNO controller using a PCRep message (labeled as 5 in Fig. 2), which eventually informs the NMS.

V. MP-AFRO VALIDATION

A. MP-AFRO Performance Evaluation

To evaluate the performance of MP-AFRO, we use two representative core network topologies: the 30-node Spanish Telefonica (TEL) and the 28-node European (EON) networks. We consider the fiber links with the spectrum width equal to 2 THz, divided into frequency slices of 12.5 GHz and granularity 6.25 GHz.

The MP-AFRO heuristic was developed in C++ and integrated in an ad-hoc event driven simulator based on OMNET++. Connection requests are generated following a Poisson process and are torn down after an exponentially distributed holding time with mean equal to 6 months. The source and destination nodes are randomly chosen using the uniform distribution.

As in [21], the bitrate of each connection request was randomly selected considering 80% of the connections being 100 Gb/s and the other 20% of 400 Gb/s. To convert bitrate into spectrum width, we use the correspondence in Table IV. Finally, note that each point in the results is the average of 10 runs with 100,000 connection requests each. Finally, we assume a link failure rate of $2.72 \cdot 10^{-3}$ per km per year [22] and consider that the link is repaired immediately after the restoration process has ended.

Aiming at comparing the performance when the MP-AFRO is used, Fig. 3 plots the blocking probability (P_b) against the offered load with and without applying MP-AFRO. P_b is weighted using connections' bitrate and the load is normalized to the largest load unleashing $P_b < 4\%$.

Although the actual load gain depends, among other factors, on the characteristics of network topology considered, gains above 13% can be obtained using MP-AFRO.

Once the performance of MP-AFRO has been evaluated, the proposed workflow is experimentally validated next.

B. Experimental Assessment

For the experiments, we consider the network topology illustrated in Fig. 1b. The data plane includes programmable spectrum selective switches (SSS) and node emulators are additionally deployed to complete the topology. Nodes are handled by co-located controllers running RSVP-TE with flexgrid extensions. The controllers run a proprietary configuration tool for automatic filter re-shaping with a resolution of 1GHz. Controllers communicate with the fPCE by means of PCEP through Gigabit Ethernet interfaces.

Fig. 4 depicts the distributed field trial set-up connecting premises in Telefonica (Madrid, Spain), CNIT (Pisa, Italy), and UPC (Barcelona, Spain) through IPsec tunnels where experiments have been carried out. Telefonica's ABNO controller was implemented in Java [14], CNIT's fPCE as well as UPC's bPCE were implemented in C++ for Linux. All components communicate by exchanging PCEP messages.

A link failure triggered a multipath restoration algorithm that rerouted the failed connections splitting them into multiple sub-connections per demand reducing its spectral efficiency and consuming higher amount of optical resources. After the failed link has been repaired, the NMS operator decided to trigger the MP-AFRO algorithm to restore the network efficiency, so the NMS requests the computation to the ABNO controller (IP: 172.16.104.2), who in turn sends the PCReq message (message 1 in Fig. 4a) to the fPCE (172.16.101.3) containing the failed link identification by means of an IRO object, and an OF object to specify the algorithm to be executed, in this experiment the MP-AFRO.

When the fPCE receives the ABNO request, it runs the *Find Candidate Demands* algorithm using as repaired link the one encoded into the received IRO object, to obtain the

candidate list of demands to be re-optimized, and a PCReq to be sent to the bPCE (IP: 172.16.50.2) is composed.

This PCReq message encodes the candidate demands using RP objects to identify them, an END-POINTS object defining its source and target nodes, a BANDWIDTH object to specify the requested bandwidth, and one RRO object for each sub-connection being used by the demand. Included candidate demands are grouped by means of a SVEC object so that they are re-optimized jointly. The received OF and IRO objects are also included in the PCReq message to inform to the bPCE on which algorithm should be executed and which link has been repaired respectively (message 2).

Fig. 4b details the PCReq message received by the bPCE highlighting the RRO objects encoding the original sub-connections. The first RRO object and one of the Label Control sub-objects has been expanded to identify its original sub-connection's route and frequency slot. Objects in Fig. 4b correspond to demand $P2a$ and $P2b$ in Fig. 1b, composed by two sub-connections. $P2a$ traverses nodes 8-9-10 using frequency slot $\langle n=2, m=2 \rangle$, and $P2b$ traverses nodes 8-1-2-3-4-5-10 using frequency slot $\langle n=1, m=1 \rangle$.

After receiving the PCReq message, the bPCE computes the MP-AFRO algorithm and responds using a PCRep message (message 3) containing the solution found. For each demand, the received RP object is duplicated and included into the PCRep message, adding also a BANDWIDTH object specifying the served bandwidth, and one ERO object for each sub-connection to be established to serve the demand. The ERO objects have the same format than the RRO objects.

Fig. 4c details the PCRep message replied by the bPCE highlighting the ERO objects encoding the re-optimized sub-connections. In contrast to Fig. 4b, objects in Fig. 4c correspond to demand $P2$, the result of merging $P2a$ and $P2b$, in Fig. 1b. The new computed connection traverses nodes 8-6-7-10 using frequency slot $\langle n=4, m=4 \rangle$. The resulting connection has restored the initially failed connection using the repaired link.

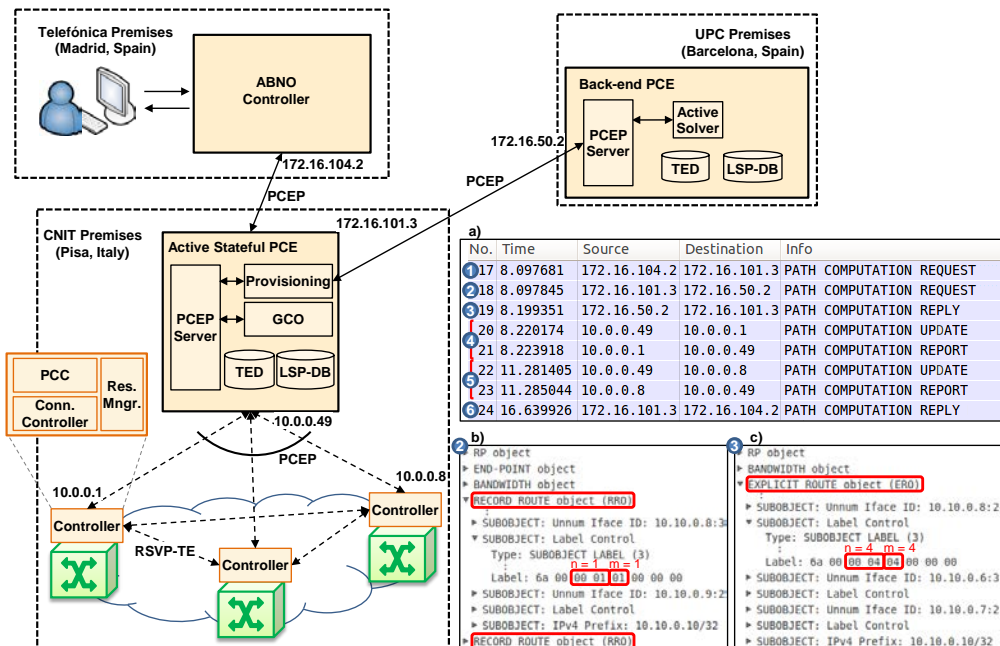


Fig. 4. Distributed field trial set-up and exchanged messages.

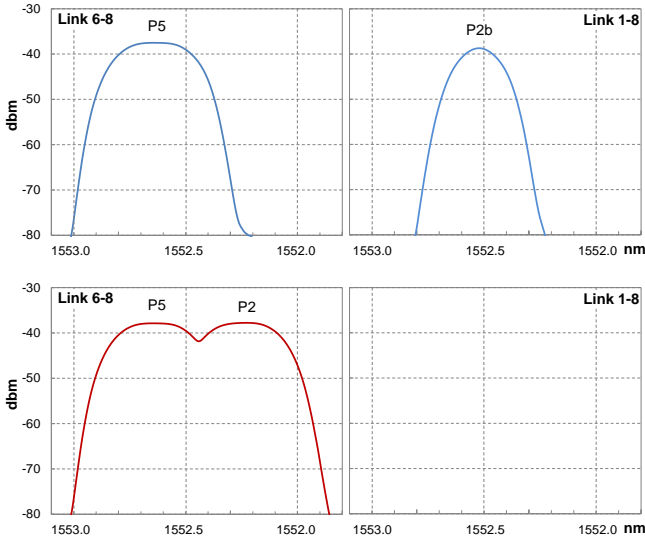


Fig. 5. Optical spectrum before (top) and after MP-AFRO (bottom) in links 6-8 and 1-8.

Upon PCRep message arrival, the fPCE starts updating connections by sending PCUpd messages towards the involved GMPLS controllers (IPs: 10.0.0.X). Data plane nodes are located in network 10.10.0.X. The controllers operate on a flexgrid test-bed derived from [23], including 100 Gb/s polarization multiplexed quadrature phase shift keying (PM-QPSK) optical signals and bandwidth variable cross-connects. Fig. 5 shows the optical spectrum on links 6-8 and 1-8 before and after the re-optimization is performed. PCRpt responses are generated when the requested actions have been performed (messages 4-5). When the solution computed by the bPCE has been completely deployed, the fPCE confirms the ABNO controller the end of the requested re-optimization by sending a PCRep message (6).

The overall re-optimization was successfully completed in around eight seconds, including circa 150 ms of pure control plane contribution (i.e., messages exchange and path computation algorithms at fPCE and bPCE). Note that around 1 s is required to tune each SSS.

VI. CONCLUSIONS

In this paper, we faced a use case of in-operation network planning. When a link fails, multipath restoration can be used to increase restorability of affected connections at the cost of worse resource utilization and spectral efficiency. After the link is repaired, the multipath after failure repair optimization (MP-AFRO) problem can be used to aggregate multiple sub-connections serving a single demand using shorter routes, thus releasing spectrum resources that now can be used to convey new connection requests. The MP-AFRO was modelled as using a MILP formulation and a heuristic algorithm was devised to find good feasible solutions in practical computation times.

After evaluating its performance on an ad-hoc network simulator, this use case of in-operation network planning was experimentally demonstrated for the first time on a distributed test-bed connecting premises in Telefonica, CNIT, and UPC. After a link was repaired, network re-optimization was requested from the NMS. The ABNO architecture controlled a flexgrid-based optical network, where the PCE architecture consisted of an fPCE and a bPCE.

The ABNO controller is in charge of initiating the MP-AFRO workflow, requesting re-optimization to the fPCE, which delegates complex computations to the bPCE. The relevant PCEP messages were shown and its contents analyzed. Note that since network dynamicity frequently derives into not optimal use of resources, this use case can easily be extended to be triggered by any other event.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 238875 GÉANT (open call REACTION) project.

REFERENCES

- [1] L. Velasco, D. King, O. Gerstel, R. Casellas, A. Castro, and V. López, "In-Operation Network Planning," *IEEE Comm Mag.*, vol. 52, pp. 52-60, 2014.
- [2] A. Farrel, J. Vasseur, J. Ash, "A Path Computation Element (PCE)-Based Architecture," *IETF RFC 4655*, 2006.
- [3] Open Networking Foundation. <https://www.opennetworking.org/>
- [4] L. Velasco, A. Castro, M. Ruiz, and G. Junyent, "Solving Routing and Spectrum Allocation Related Optimization Problems: from Off-Line to In-Operation Flexgrid Network Planning," *IEEE J. of Lightwave Technol.*, vol. 32, pp. 2780-2795, 2014.
- [5] S. Ma et al., "Demonstration of online spectrum defragmentation enabled by OpenFlow in software-defined elastic optical networks," in *Proc. OFC*, 2014.
- [6] A. Castro et al., "Experimental Demonstration of an Active Stateful PCE Performing Elastic Operations and Hitless Defragmentation," in *Proc. ECOC*, 2013.
- [7] D. King, and A. Farrel, "A PCE-based Architecture for Application-based Network Operations," *IETF draft*, work in progress, 2014.
- [8] E. Crabbe, J. Medved, I. Minei, and R. Varga, "PCEP Extensions for Stateful PCE," *IETF draft*, work in progress, 2014.
- [9] E. Crabbe, I. Minei, S. Sivabalan, and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model," *IETF draft*, work in progress, 2014.
- [10] L. Gifre, L. Velasco, N. Navarro, and G. Junyent, "Experimental Assessment of a High Performance Back-end PCE for Flexgrid Optical Network Re-optimization," in *Proc. OFC*, 2014.
- [11] J. Vasseur, and J. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)," *IETF RFC 5440*, 2009.
- [12] H. Gredler, J. Medved, S. Previdi, A. Farrel, S. Ray, "North-Bound Distribution of Link-State and TE Information using BGP," *IETF draft*, work in progress, 2014.
- [13] E. Crabbe, J. Medved, I. Minei, R. Varga, "PCEP Extensions for Stateful PCE," *IETF draft*, work in progress, 2014.
- [14] A. Aguado et al., "ABNO: a feasible SDN approach for multi-vendor IP and optical networks," in *Proc. OFC*, 2014.
- [15] L. Gifre, F. Paolucci, A. Aguado, R. Casellas, A. Castro, F. Cugini, P. Castoldi, L. Velasco, and V. López, "Experimental Assessment of In-Operation Spectrum Defragmentation," *Photonic Network Communications*, vol. 27, pp. 128-140, 2014.
- [16] M. Ruiz, M. Zolkiewicz, A. Castro, M. Klinkowski, L. Velasco, and M. Pioro, "After Failure Repair Optimization in Dynamic Flexgrid Optical Networks," in *Proc. OFC*, 2014.
- [17] A. Castro, L. Velasco, J. Comellas, and G. Junyent, "On the benefits of Multi-path Recovery in Flexgrid Optical Networks," *Photonic Network Communications*, 2014.
- [18] L. Velasco et al., "First Experimental Demonstration of ABNO-driven In-Operation Flexgrid Network Re-Optimization," *PDP OFC*, 2014.
- [19] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow, "Extensions to RSVP for LSP Tunnels," *IETF RFC 3209*, 2001.
- [20] R. Ramaswami, K. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE Trans. on Netw.*, vol. 3, pp. 489-500, 1995.
- [21] F. Paolucci, A. Castro, F. Cugini, L. Velasco, and P. Castoldi, "Multipath Restoration and Bitrate Squeezing in SDN-based Elastic Optical Networks," *Photonic Network Communications*, vol. 28, pp. 45-57, 2014.
- [22] W. D. Grover, "Mesh-Based Survivable Networks," Prentice Hall PTR, New Jersey, 2004.
- [23] F. Cugini et al., "Demonstration of Flexible Optical Network Based on Path Computation Element," *IEEE J. of Lightwave Technol.*, vol. 30, pp. 727-733, 2012.